

PLC

Skuteczne programowanie

Niniejszy kurs opisuje sposób korzystania z programowania strukturalnego, aby zwiększyć skuteczność programowania. Kurs jest skierowany do osób programujących przy użyciu prostych schematów drabinkowych.

W ramach tego kursu poznasz wszystkie zalety programowania strukturalnego. Mamy nadzieję, że kurs pomoże w bardziej skutecznym programowaniu.

Warunkiem wstępnym przystąpienia do niniejszego kursu jest wcześniejsze ukończenie poniższych kursów:

- Programming Basic (Programowanie podstawowe)

Treść tego kursu posiada następującą strukturę.

Rozdział 1 – Dlaczego programowanie strukturalne jest potrzebne?

Rozdział ten opisuje konieczność stosowania programowania strukturalnego w celu uzyskania większej skuteczności.

Rozdział 2 – Dlaczego programy strukturalne są takie skuteczne

Rozdział ten opisuje zalety programowania strukturalnego.

Test końcowy

Ocena zaliczająca: 60% lub więcej

Przejdź do następnej strony		Przejdź do następnej strony.
Wróć do poprzedniej strony		Wróć do poprzedniej strony.
Przejdź do żądanej strony		Wyświetli się „Spis treści” umożliwiający przejście do żądanej strony.
Zakończ naukę		Zakończ naukę.

Zalecenia dotyczące bezpieczeństwa

Jeśli uczysz się, korzystając z rzeczywistych produktów, prosimy o dokładne przeczytanie zasad bezpieczeństwa zawartych w odpowiednich instrukcjach obsługi.

Środki ostrożności dla tego kursu

Ekrany wyświetlane dla wersji oprogramowania, którego używasz, mogą się różnić od przedstawionych w tym kursie. W ramach tego kursu użyto następującej wersji oprogramowania:

- GX Works3 wersja 1.032J

Rozdział 1 Dlaczego programowanie strukturalne jest potrzebne?

Rozdział ten opisuje konieczność stosowania programowania strukturalnego do bardziej skutecznej inżynierii.

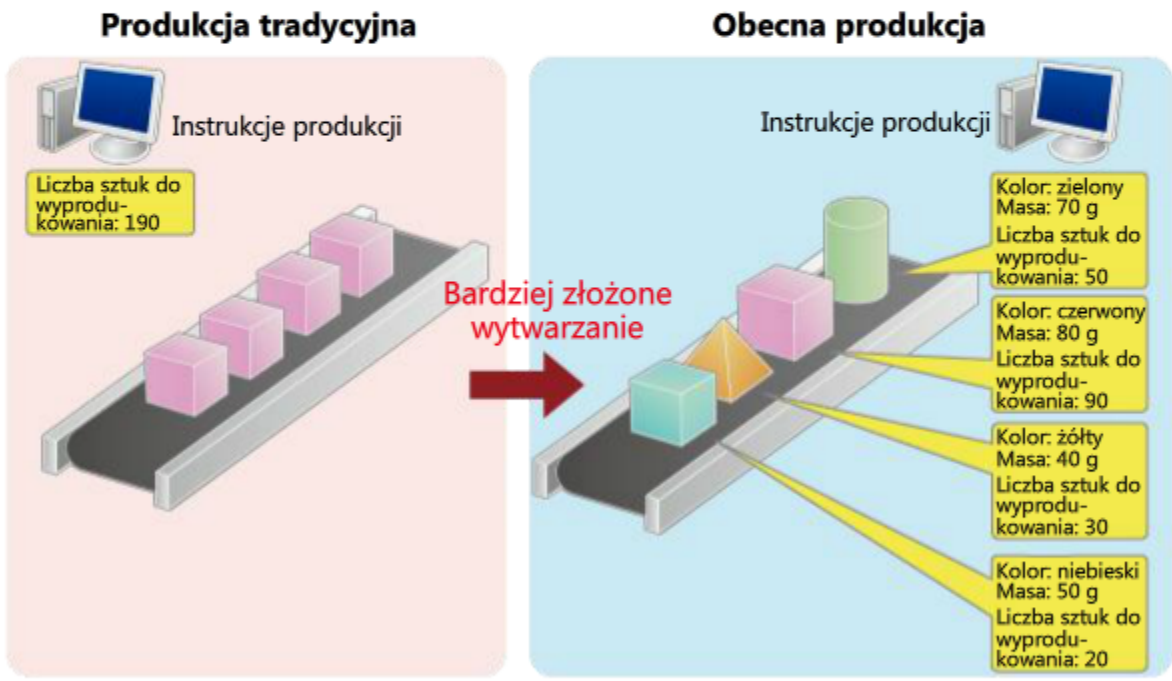
- 1.1 Stan obecny w branży FA
- 1.2 Konieczność uzyskania skuteczności programowania
- 1.3 Skuteczne programowanie dzięki programom strukturalnym
- 1.4 Programowanie strukturalne przy użyciu MELSOFT GX Works3

1.1 Stan obecny w branży FA

Obecnie w branży FA ilość danych przetwarzanych przez urządzenia zwiększa się wraz ze wzrostem stopnia zaawansowania i złożoności procesów produkcyjnych. Szczegółowe informacje zostały opisane poniżej.

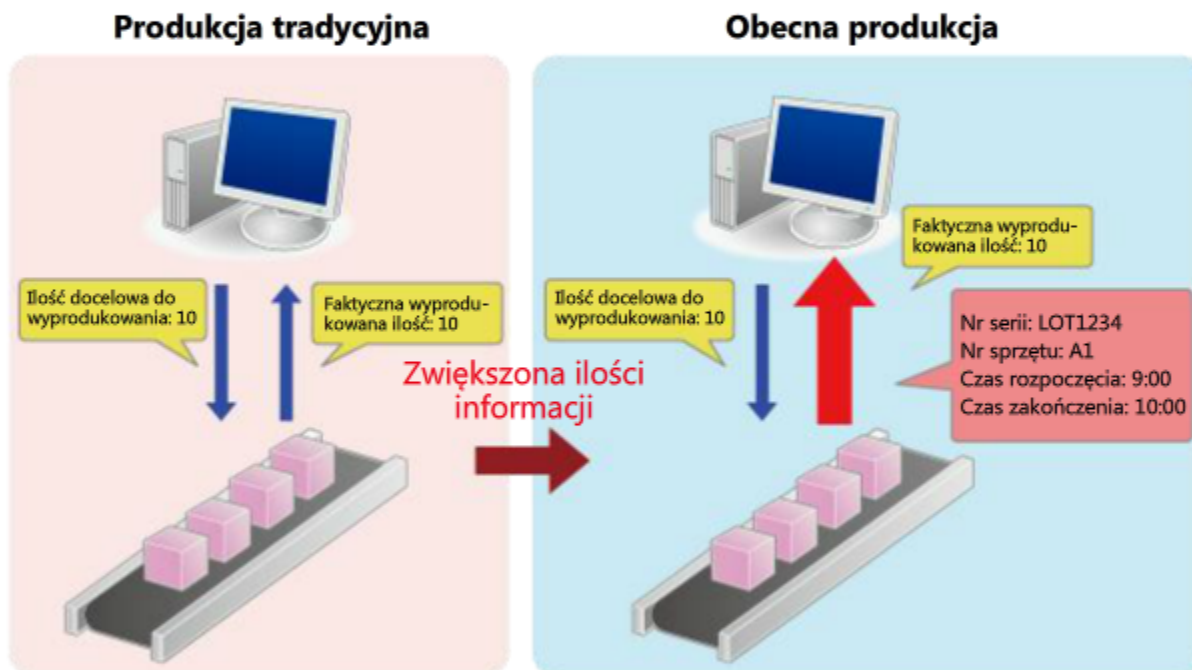
1. Bardziej złożone przetwarzanie

Wcześniej pojedynczy produkt był wytwarzany na jednej linii produkcyjnej, zgodnie z jedną instrukcją produkcji. Wytwarzanie wielu produktów na jednej linii produkcyjnej staje się coraz bardziej powszechne w wyniku zróżnicowania produktów oraz uproszczenia systemów produkcji. Z tego względu produkcja stała się bardziej złożona.



2. Zwiększenie ilości informacji

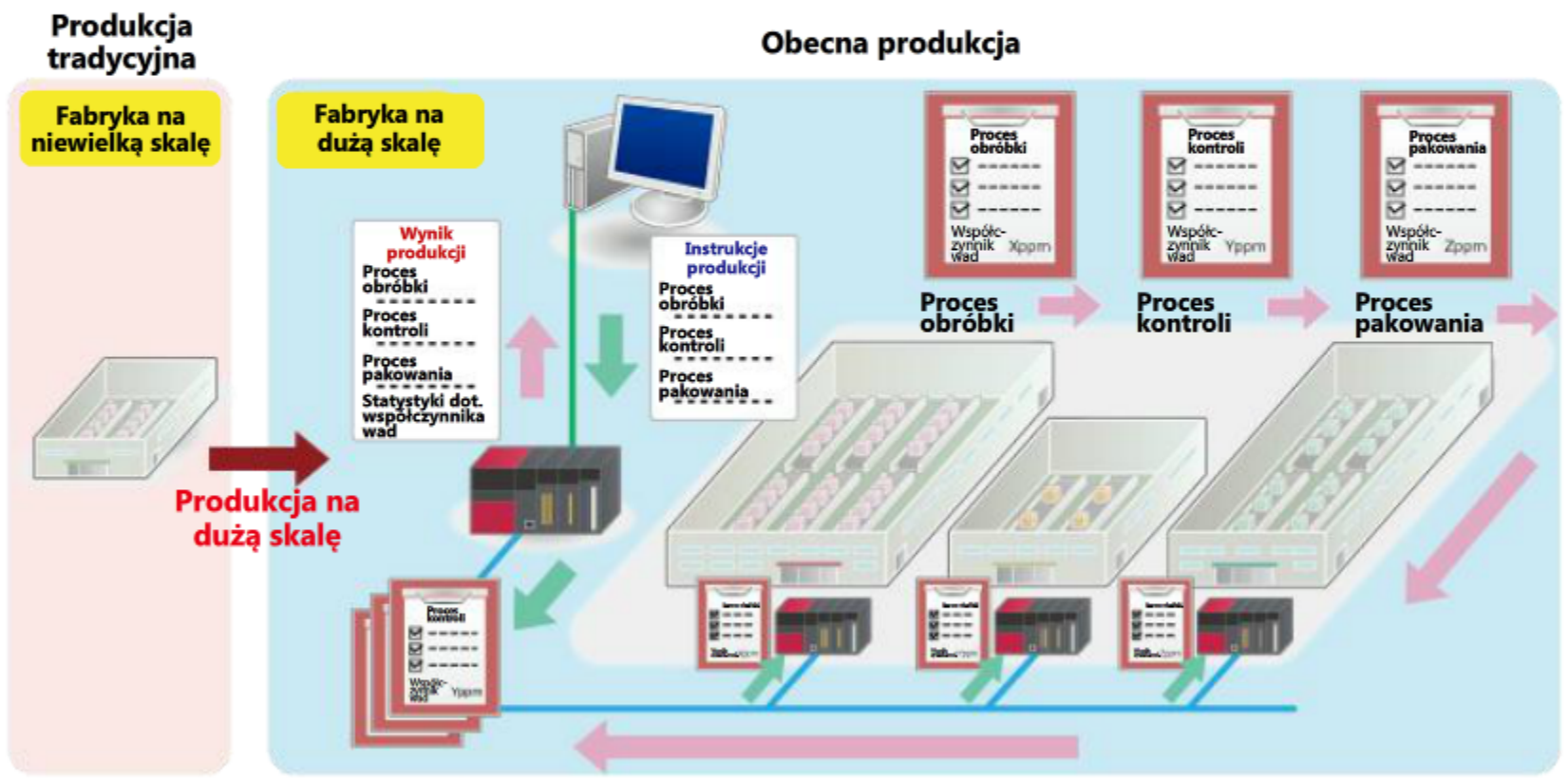
Wcześniej ilość informacji wymienianych pomiędzy komputerami na poziomie przedsiębiorstwa a sprzętem linii produkcyjnej była ograniczona do instrukcji pojedynczego zadania oraz wyników produkcji. Obecnie parametry produkcji są uwzględniane w informacjach obejmujących sterowanie produkcją w celu zapewnienia możliwości ich śledzenia. Ilość informacji przetwarzanych w zakładach produkcyjnych wzrosła.



1.1 Stan obecny w branży FA

3. Konfiguracja systemu na dużą skalę

Budowa i rozwój fabryk na dużą skalę stają się coraz bardziej powszechne. W tych fabrykach sieci są skonfigurowane pod kątem każdego procesu produkcji a informacje z każdego procesu są gromadzone przez komputery na poziomie przedsiębiorstwa. Ilość informacji przetwarzanych w tych środowiskach jest znacząco większa niż ilość w tradycyjnych fabrykach na niewielką skalę.



1.1

Stan obecny w branży FA

Wraz ze wzrostem ilości informacji wykorzystywanych w środowisku FA, zmienił się również zakres pracy sterowników programowalnych.

Wcześniej sterowniki były głównie stosowane do sterowania, podczas gdy komputery służyły do przetwarzania informacji. Zarówno ilość przechowywanych danych oraz prędkość przetwarzania sterowników programowalnych wzrosła. Sterowniki programowalne zostały zaprojektowane do pracy w środowiskach FA i z tego względu są bardziej niezawodne i odporne niż komputery. Dlatego zaobserwowano tendencję polegania na nich podczas przetwarzania informacji.

Produkcja tradycyjna

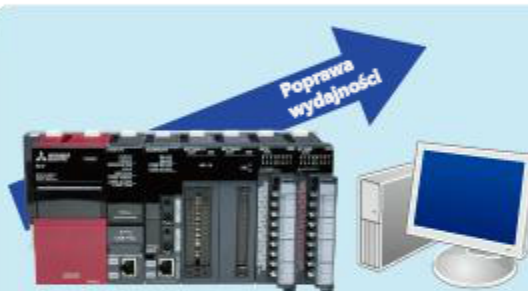


Rozwój zakresu przetwarzania przez sterowniki programowalne

Sterowanie

Przetwarzanie informacji

Obecna produkcja



Sterowanie

Przetwarzanie informacji

Przetwarzanie przy użyciu sterownika programowalnego było coraz bardziej powszechne, programy sterujące są obecnie stosowane do bardziej złożonego przetwarzania oprócz sterowania sekwencyjnego, a liczba kroków wzrosła. Przykłady obejmują przetwarzanie danych tekstowych, wykonywanie operacji na bazach danych oraz sterowanie siecią.

W tradycyjnych programach stworzonych przy użyciu prostych schematów drabinkowych wszystkie funkcje przetwarzania są opisane bez wyraźnych różnic. Złożone programy obejmujące wiele kroków sprawiają duże problemy w przypadku konieczności edycji, lub późniejszego dodania nowej funkcji.

Wykorzystanie istniejących programów nie jest łatwe



Określenie początku lub końca przetwarzania jest trudne.

Należy wyszukać części nadające się do wykorzystania, odnosząc się do komentarzy i wyjaśnień.



W miejscach opracowywania oprogramowania, wydajne programowanie poprzez np. wielokrotne wykorzystywanie takich samych programów i ich udostępnianie (standaryzacja) skraca czas opracowania nowych projektów.

1.3 Skuteczne programowanie dzięki programom strukturalnym

Wadą tradycyjnych, prostych programów drabinkowych jest fakt, że są one trudne do zrozumienia i wykorzystania. Programy strukturalne mogą zwiększyć skuteczność programowania i wyeliminować te problemy.

Poniższa animacja opisuje koncepcję podziału programu na części.

Programy można wykorzystać, wybierając części pojedynczego programu.




Istniejący program



Nowy program wykorzystujący istniejące programy



Wystarczy tylko wybrać część programu

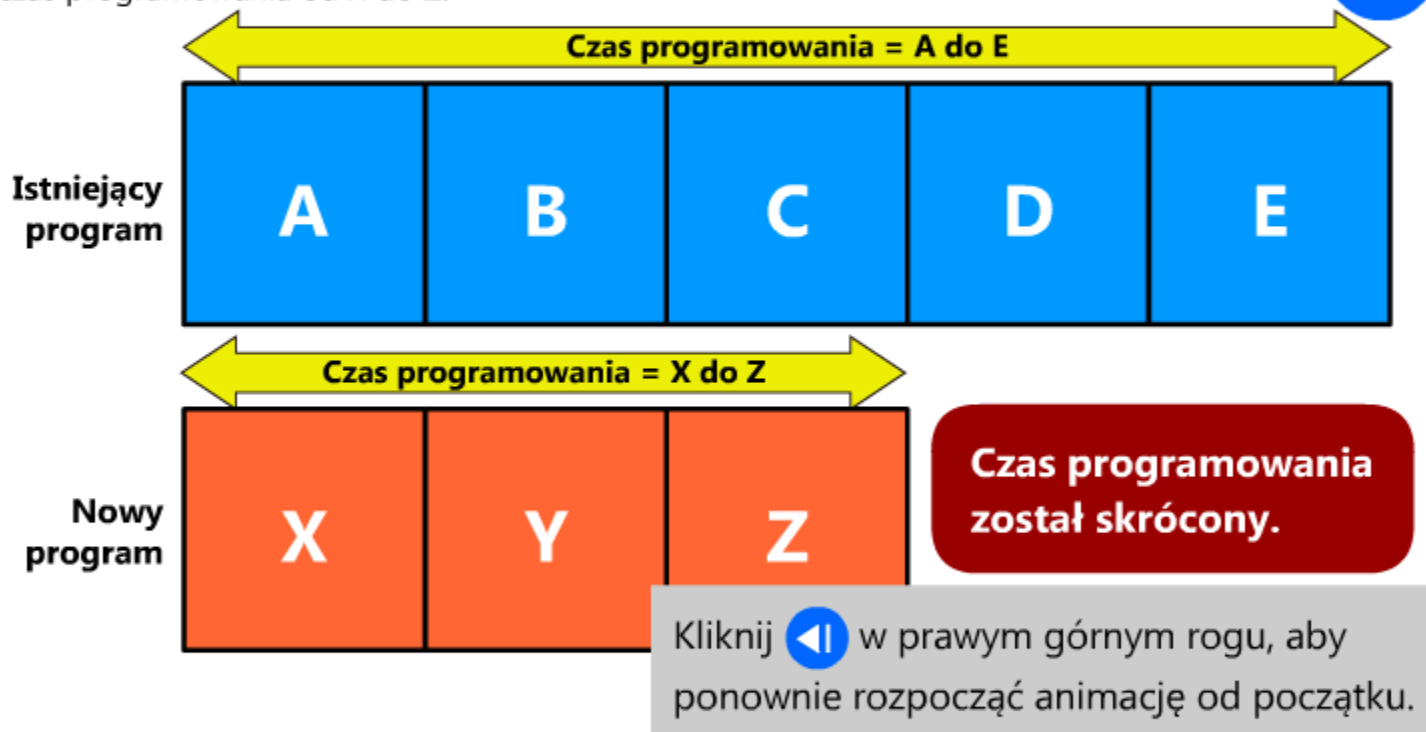
Kliknij  w prawym górnym rogu, aby ponownie rozpocząć animację od początku.

Wykorzystanie programów zwiększa skuteczność programowania. Szczegółowe informacje zostały opisane w następnych częściach.

1.3.1 Korzyści wykorzystania programów

Wykorzystanie istniejących programów w maksymalnym stopniu skraca czas programowania.

Nowy program obejmuje przetwarzanie przedstawione jako B i E w istniejącym programie. Przetwarzanie to można wykorzystać, więc do utworzenia nowego programu wymagany jest czas programowania od X do Z.



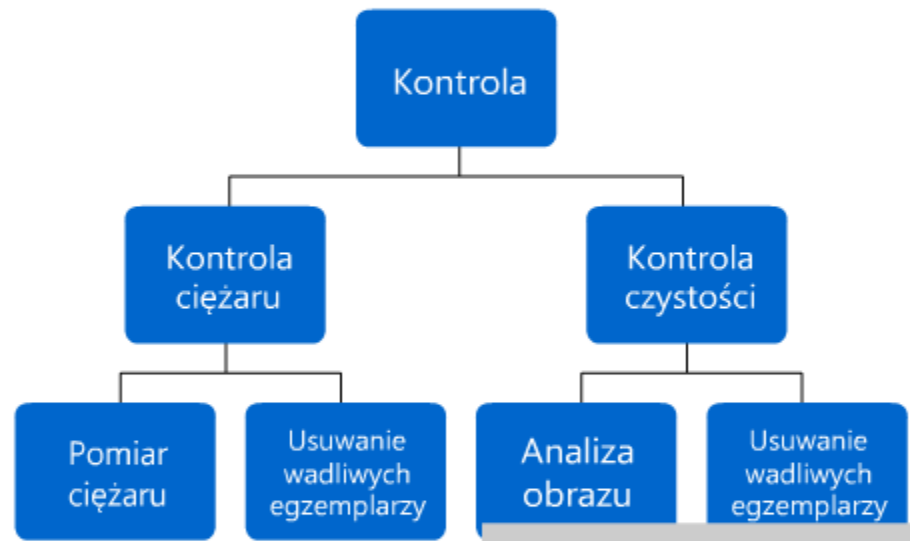
W ten sposób wykorzystanie istniejących programów wyeliminuje konieczność pisania ich od nowa, skracając ogólny czas programowania. Maksymalne wykorzystanie istniejących programów znacząco zwiększa skuteczność programowania.


Podstawowa koncepcja tworzenia programów strukturalnych została opisana w kolejnych częściach.

1.3.2 Skuteczny wybór części nadających się do wykorzystania

Poniższa animacja opisuje koncepcję organizacji hierarchicznej. Zastosowanie organizacji hierarchicznej umożliwia skuteczniejszy wybór części nadających się do wykorzystania.

Ułożenie POU w ten sposób umożliwia wyświetlenie programów w mniejszych częściach oraz na wyższym poziomie jednocześnie, dzięki czemu można bardziej skutecznie wybrać części nadające się do wykorzystania.



Kliknij  w prawym górnym rogu, aby ponownie rozpocząć animację od początku.

1.3.2

Skuteczny wybór części nadających się do wykorzystania


Ten przykład ilustruje, jak skuteczne może być wykorzystanie części programu zorganizowanego hierarchicznie.

Istniejący program

Nowy program wykorzystujący istniejące programy



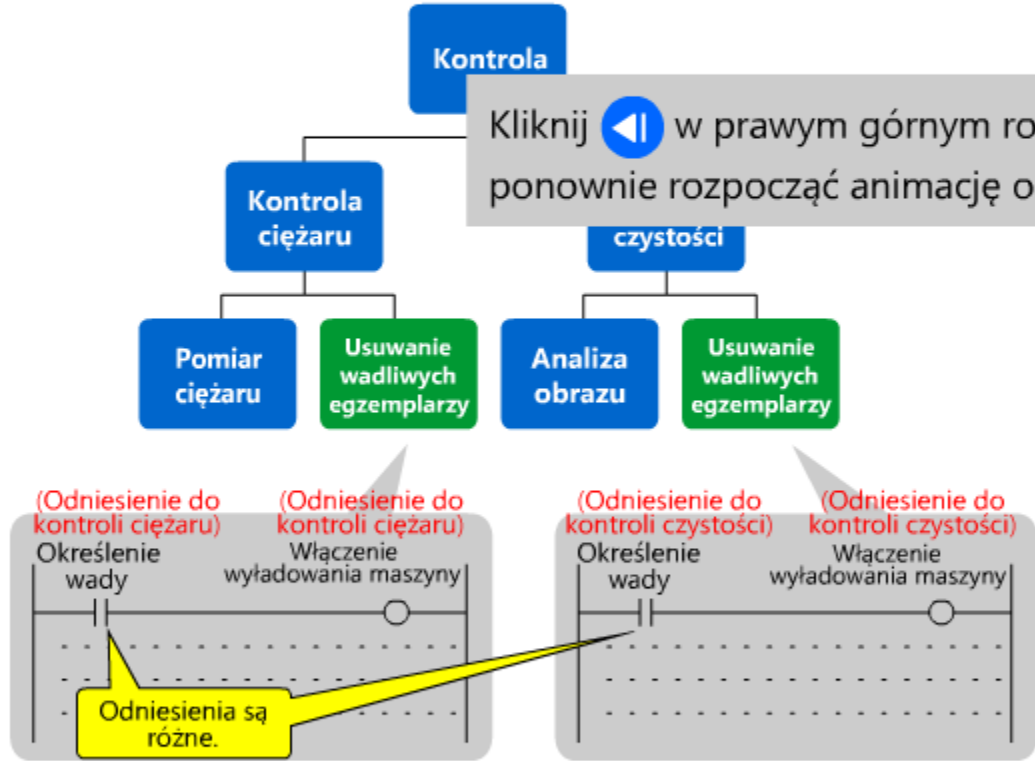
Wykorzystanie wielu części

Kliknij  w prawym górnym rogu, aby ponownie rozpocząć animację od początku.

1.3.3 Wykorzystanie programów bez konieczności ponownego przypisania urządzeń

Przedstawiliśmy sposób, w jaki programowanie strukturalne ułatwia wykorzystanie programów. Dzięki programowaniu strukturalnemu w miejscu urządzeń stosowane są etykiety. Stosowanie etykiet umożliwia programistom tworzenie programów bez martwienia się konfliktem urządzeń z nowym programem.

Kontrola duplikatów w przypadku urządzeń nie jest wymagana.



Kliknij w prawym górnym rogu, aby ponownie rozpocząć animację od początku.

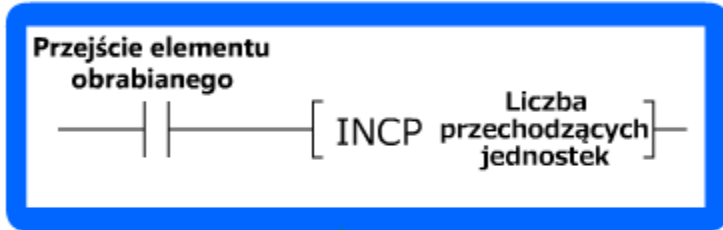
1.3.4 Programowanie przy użyciu etykiet do określonego przetwarzania

Stosowanie etykiet umożliwia tworzenie programów bez konieczności stosowania komentarzy i fizycznych adresów.


Wykorzystanie etykiet umożliwia tworzenie programów z nazwą przetwarzania bez konieczności martwienia się o urządzenia.



Wykorzystanie etykiet



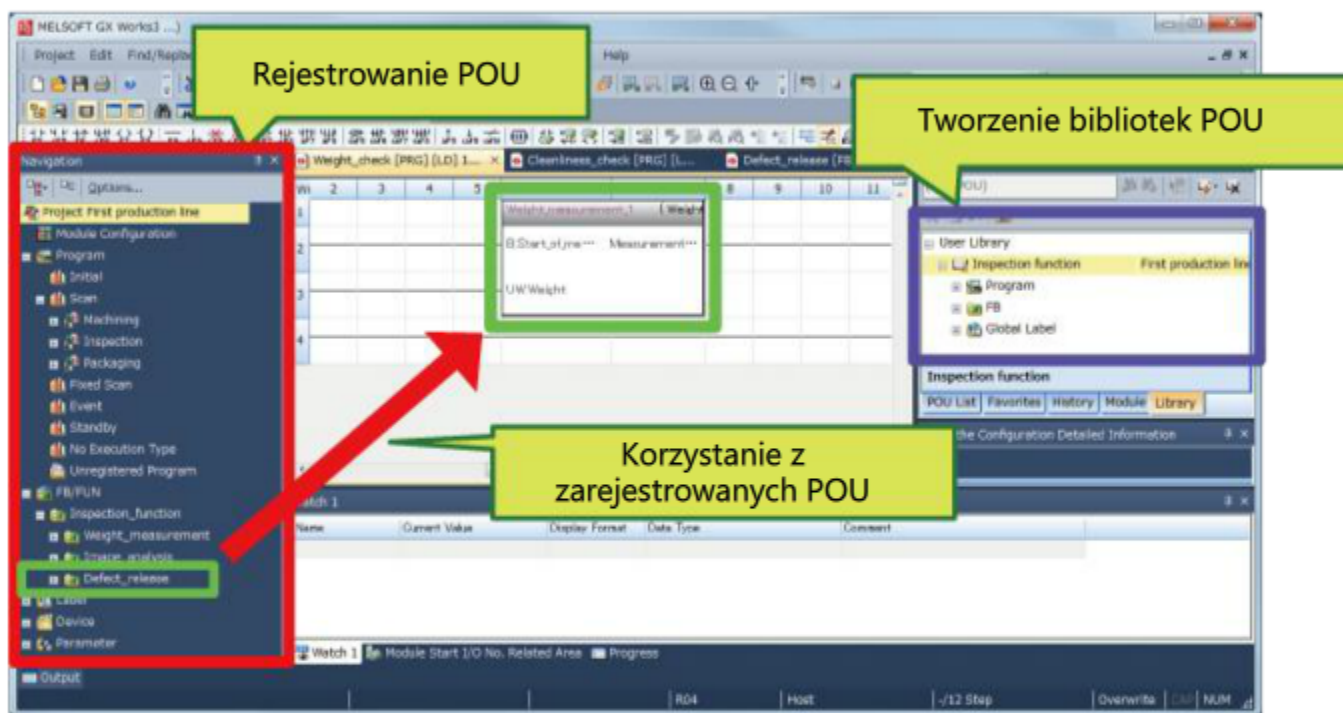
Intuicyjne i łatwe w zrozumieniu

Kliknij  w prawym górnym rogu, aby ponownie rozpocząć animację od początku.

1.4 Programowanie strukturalne przy użyciu MELSOFT GX Works3

Oprogramowanie inżynierskie MELSOFT GX Works3 umożliwia tworzenie programów strukturalnych.

MELSOFT GX Works3 ułatwia wykorzystanie programów, wyświetlając dostępne jednostki organizacyjne programu POU (program organization units) w celu łatwego zarządzania.



MELSOFT GX Works3

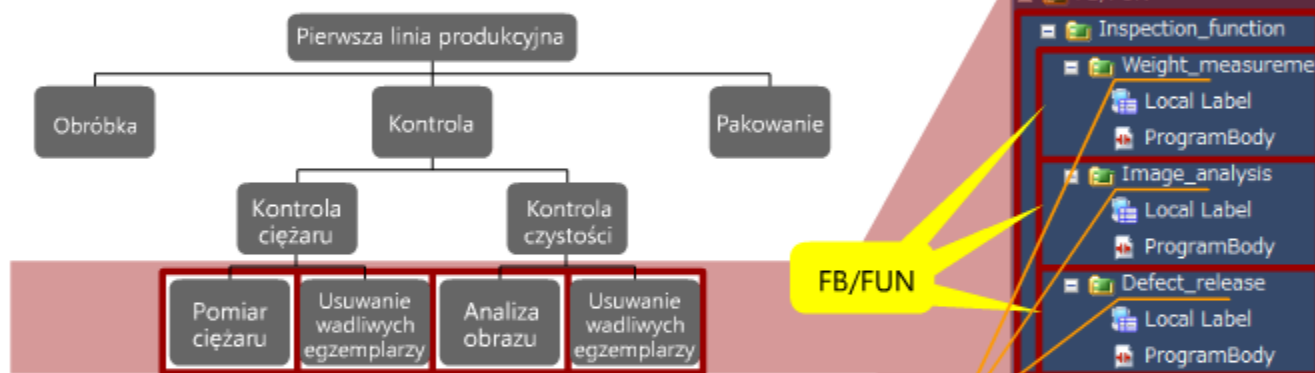
1.4.1

Organizacja hierarchii programów w GX Works3

Poniższa animacja opisuje sposób przedstawienia organizacji hierarchicznej programów w GX Works3 przy użyciu przykładowej linii produkcyjnej.


Programy do kontroli ciężaru obejmują funkcję pomiaru ciężaru i zwalniania wad.

Funkcja pomiaru ciężaru, analizy obrazu i usuwania wadliwych egzemplarzy objęta przetwarzaniem jest przedstawiona jako niższy poziom pod poziomem FB/FUN.



Funkcja

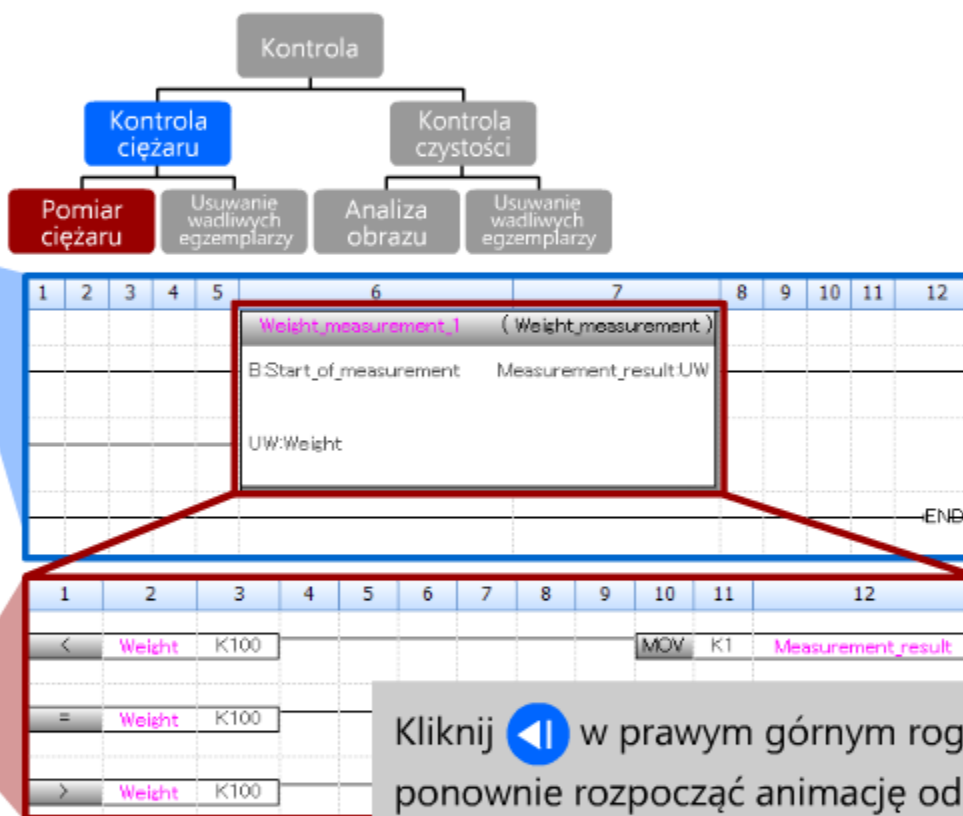
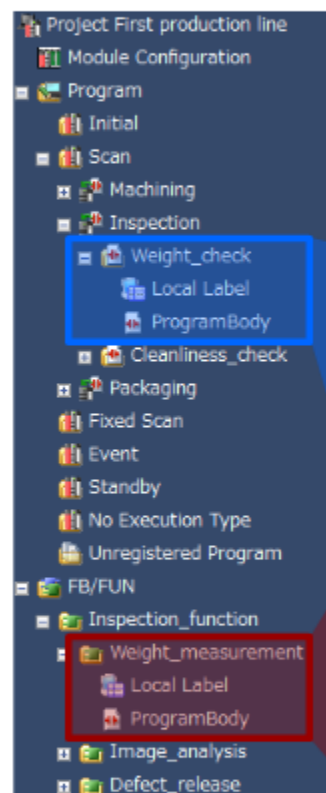
Nazwy FB oraz FUN może przypisywać użytkownik.


Kliknij  w prawym górnym rogu, aby ponownie rozpocząć animację od początku.

1.4.2

Korzystanie z jednostek organizacyjnych programu POU (program organization units)

Ta część wyjaśnia procedurę stosowania POU w GX Works3. FB/FUN zostały wybrane z okna projektu i zorganizowane w ramach programu.



Kliknij  w prawym górnym rogu, aby ponownie rozpocząć animację od początku.

1.4.3 Korzystanie z bibliotek

POU można udostępniać w ramach projektów przy użyciu funkcji biblioteki. Tworzenie bibliotek sprawdzonych POU gwarantuje stałą jakość programów.

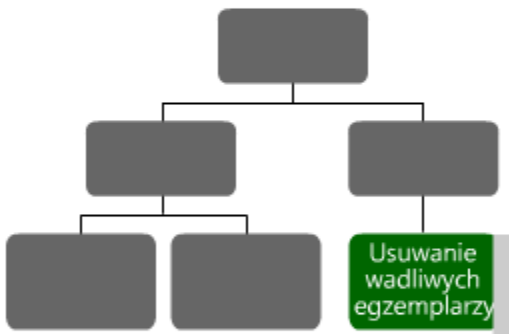
POU można stosować poprzez proste przeciągnięcie i upuszczenie w programie.




ProgPou [PRG] [LD] 12Step x						
2	3	4	5	6	7	
				Defect_release 2	(Defect_release)	

User Library

- Inspection function First production line
- Program
- FB
 - Defect_release
 - Image_analysis
 - Weight_measurement
- Global Label



Kliknij  w prawym górnym rogu, aby ponownie rozpocząć animację od początku.

1.4.3

Skuteczne wykorzystanie bibliotek

Szczególnie skuteczne jest tworzenie bibliotek często wykorzystywanych POU. W poniższych programach kontroli, bloki FB/FUN „Usuwanie wadliwych egzemplarzy” są najbardziej odpowiednie do rejestracji w bibliotece.



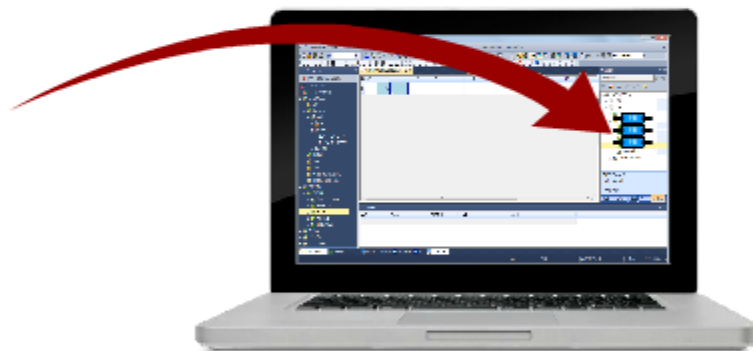
1.4.4


Wykorzystanie wstępnie opracowanych POU

GX Works3 został wstępnie skonfigurowany i zawiera POU do wielu typowych programów. Wstępnie opracowane POU zawierają bloki funkcyjne FB dla poszczególnych modułów. Są one automatycznie rejestrowane zgodnie z konfiguracją modułów w projekcie, oraz dodanymi bibliotekami MELSOFT (MELSOFT Library).

MELSOFT Library oferowane przez Mitsubishi Electric zawiera wiele przydatnych FB dostępnych dla użytkowników.

FB pomagają w stosowaniu modułów utworzonych przez Mitsubishi Electric lub produktów innych producentów.



Kliknij  w prawym górnym rogu, aby ponownie rozpocząć animację od początku.

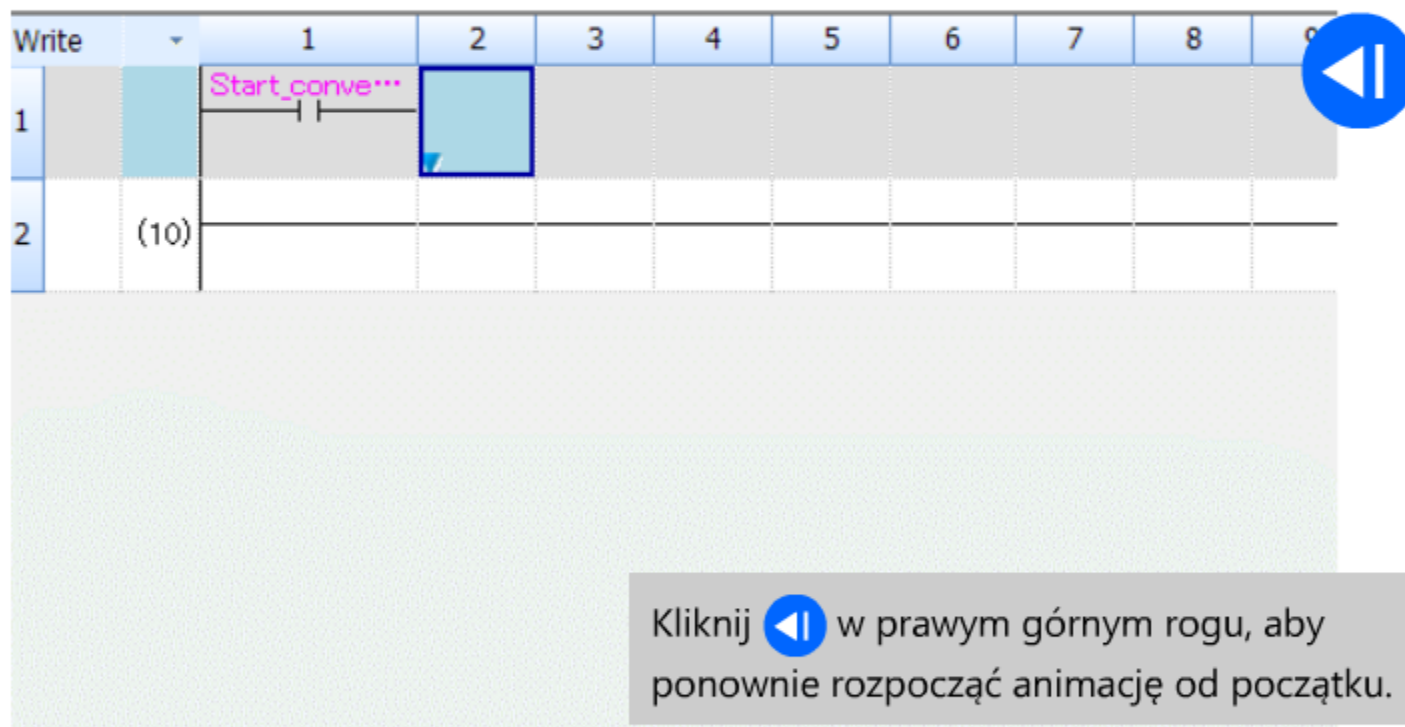
W celu uzyskania bibliotek (MELSOFT Library) należy skontaktować się z lokalnym przedstawicielem Mitsubishi Electric.

1.4.5 Skuteczne wprowadzanie etykiet


Funkcja przewidywania tekstu umożliwia bardziej skuteczne wprowadzanie etykiet.

Po rozpoczęciu wprowadzania etykiety automatycznie pojawia się lista zarejestrowanych etykiet, które pasują do pierwszych kilku wprowadzonych znaków.

Następnie pożądana etykieta jest wprowadzana poprzez jej wybranie. Umożliwia to umieszczenie tej samej etykiety w wielu miejscach bez błędów podczas wprowadzania.



Write	1	2	3	4	5	6	7	8	9
1	Start_conve...								
2	(10)								

Kliknij  w prawym górnym rogu, aby ponownie rozpocząć animację od początku.

Programowanie przy użyciu zarejestrowanych etykiet

Treść tego rozdziału to:

- Stan obecny w branży FA
- Wady programów stosujących proste drabinki
- Koncepcja i zalety programów strukturalnych
- Programowanie strukturalne przy użyciu MELSOFT GX Works3

Ważne kwestie do rozważenia:

Korzyści wykorzystania programów	Wielokrotne wykorzystanie programów zmniejsza potrzebę opracowywania nowych. Maksymalne wykorzystanie istniejących programów znacząco zwiększa skuteczność programowania.
Podział programów na części programów	Części nadające się do wykorzystania można wyodrębnić, dzieląc programy na części programów i przypisując sensowne nazwy do tych części.
Organizacja hierarchiczna	Oprócz podziału programów na części programów, korzystanie z organizacji hierarchicznej ułatwia i zwiększa skuteczność wyboru części programów nadających się do wykorzystania.
Zalety korzystania z etykiet	<ul style="list-style-type: none"> • Korzystanie z etykiet eliminuje konieczność sprawdzania konfliktu urządzeń oraz ponownego przypisywania urządzeń, co sprawia, że wykorzystanie programów jest bardziej skuteczne • Programowanie jest bardziej intuicyjne, jeśli stosowane są sensowne nazwy odzwierciedlające określone przetwarzanie
Programy strukturalne w MELSOFT	<ul style="list-style-type: none"> • MELSOFT umożliwia tworzenie programów strukturalnych w łatwym do zrozumienia edytorze wizualnym • MELSOFT pomaga zwiększyć skuteczność programowania

Rozdział 2 Dlaczego programy strukturalne są takie skuteczne

Rozdział ten opisuje zalety programowania strukturalnego.

Proces programowania zostanie opisany z podziałem na następujące etapy.

- Edytowanie programu: Tworzenie i edytowanie programu
- Test oceniający: Potwierdzenie, że program działa prawidłowo
- Wyszukiwanie i naprawianie: Wskazywanie i naprawianie błędów

2.1 Skrócenie czasu tworzenia oprogramowania poprzez wykorzystanie programów

2.2 Usunięcie procesu ponownego przypisania urządzenia

2.3 Zapobieganie przypadkowym zmianom

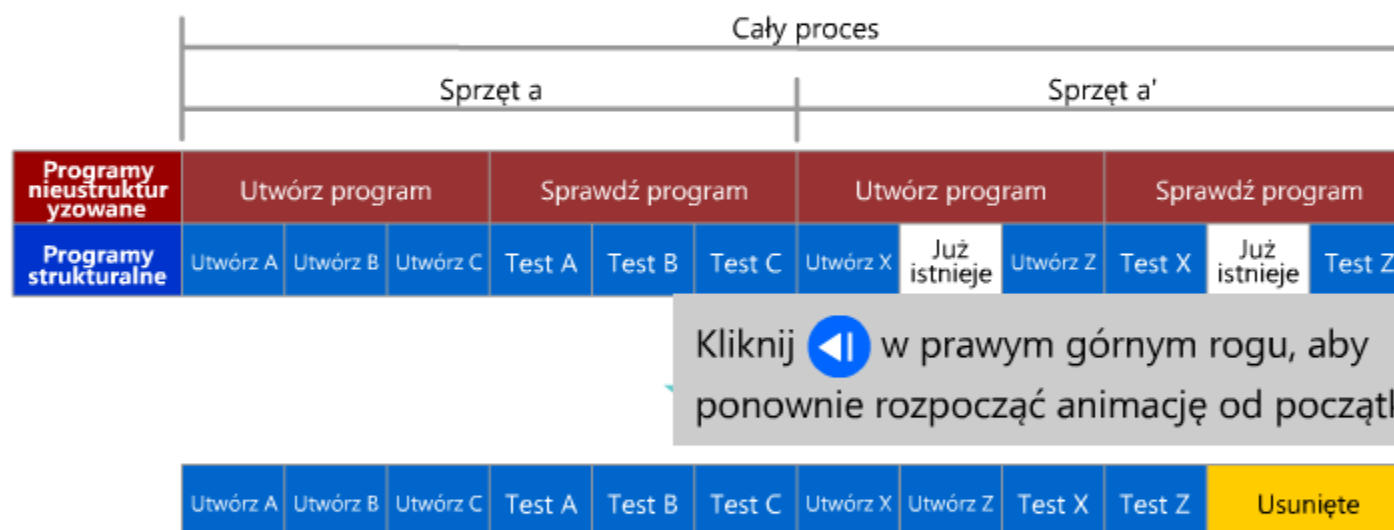
2.1 Skrócenie czasu tworzenia oprogramowania poprzez wykorzystanie programów

Wykorzystanie istniejących programów skraca czas programowania i poprawia skuteczność programowania.

Skuteczne programowanie dzięki programom strukturalnym



Proces programowania

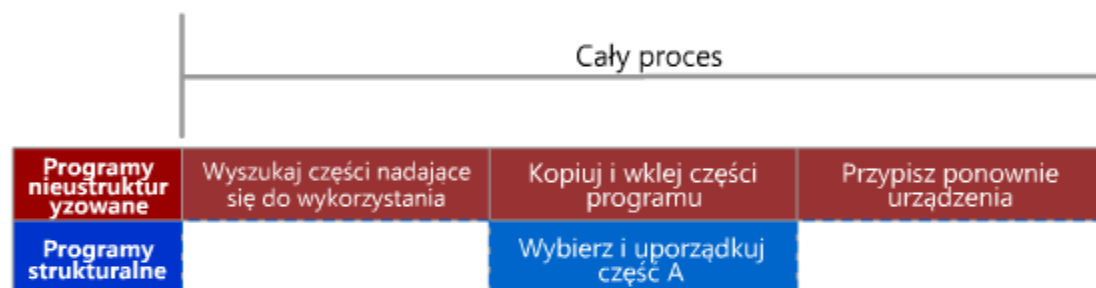



Wykorzystanie programów strukturalnych pozwala na uniknięcie konieczności ponownego przypisania urządzeń w ramach nowego programu.

Skuteczne programowanie dzięki programom strukturalnym



Proces programowania



Kliknij  w prawym górnym rogu, aby ponownie rozpocząć animację od początku.



2.3

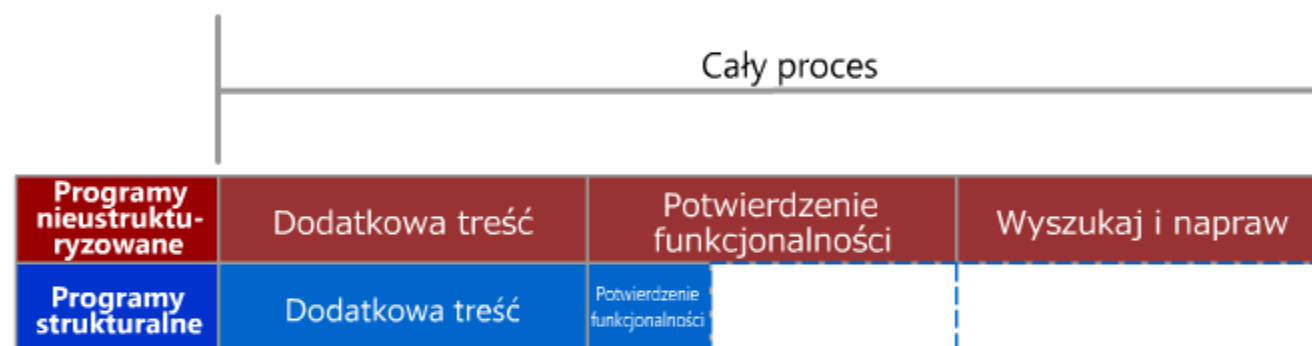
Zapobieganie przypadkowym zmianom


Poprzez zastosowanie programów strukturalnych można zapobiec przypadkowym zmianom wynikającym z dodania funkcji do programów.

Skuteczne programowanie dzięki programom strukturalnym



Proces programowania



Kliknij  w prawym górnym rogu, aby ponownie rozpocząć animację od początku.

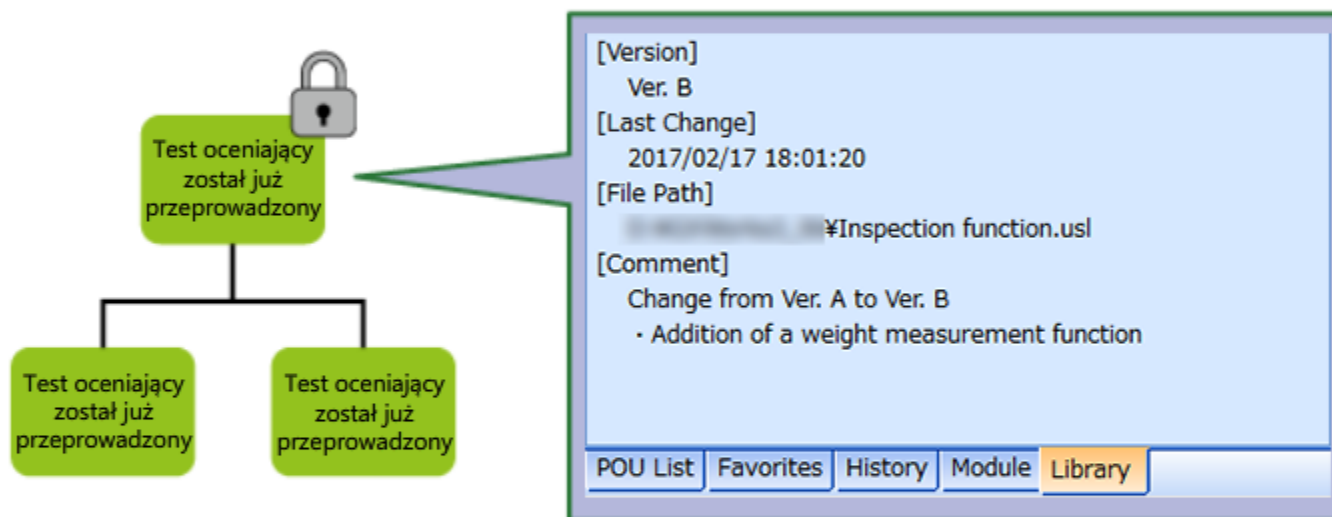


2.3.1

Bezpieczne zarządzanie POU

Przy użyciu MELSOFT GX Works3 można bezpiecznie zarządzać POU.

Poprzez zablokowanie poszczególnych POU można całkowicie wyeliminować przypadkowe zmiany. Każde POU obejmuje własne informacje, włączając wersję, datę aktualizacji oraz komentarze. Zmiana historii może zostać zapisana w komentarzach.



Treść tego rozdziału to:

- Korzyści wykorzystania POU
- Korzyści korzystania z etykiet
- Korzyści korzystania z programów strukturalnych

Ważne kwestie do rozważenia:

Zmniejszenie konieczności testów oceniających	Zastosowanie programów strukturalnych pozwala na skrócenie nie tylko czasu wymaganego do tworzenia programów, lecz także czasu niezbędnego do przetestowania oprogramowania.
Zapobieganie przypadkowym zmianom	Dzięki programom strukturalnym zmiany wprowadzone w poszczególnych POU nie mają wpływu na pozostałe POU.
Blokowanie POU	Blokowanie POU po zakończeniu testów poprawności działania zapewnia bezpieczne zarządzanie jednostkami POU.

Niniejszy kurs opisuje wykorzystanie programowania strukturalnego, do skuteczniejszego programowania. Ukończyłeś(-aś) właśnie ten kurs.

Oto podsumowanie zakresu kursu.

- Programowanie strukturalne to metoda skutecznego tworzenia programów, które stają się coraz bardziej złożone ze względu na trendy produkcji obejmującej systemy na dużą skalę oraz coraz większą ilość danych.
- Programy strukturalne są łatwiejsze w stosowaniu, co zmniejsza konieczność opracowania nowych programów i poprawia skuteczność programowania
- Podział programów na części programów umożliwia podział programów wg przetwarzania i funkcji. Używanie organizacji hierarchicznej oraz sensownych nazw części programów umożliwia wskazanie części programów, które można wykorzystać
- Zastosowanie organizacji hierarchicznej umożliwia bardziej skuteczny wybór części nadających się do wykorzystania
- Stosowanie etykiet umożliwia wykorzystanie POU bez martwienia się konfliktem urządzeń z nowym programem
- MELSOFT GX Works3 obsługuje programowanie strukturalne i zwiększa skuteczność programowania
- Tworzenie bibliotek umożliwia udostępnianie/standaryzację często stosowanych części programów, zapewniając stałą jakość programów
- Oprócz wykorzystania programów, można zastosować biblioteki w celu zmniejszenia ilości nowo opracowanych programów

W przypadku faktycznego wdrożenia programowania strukturalnego należy odnieść się do poniższych materiałów.

Programowanie przy użyciu MELSOFT GX Works3

- Kurs e-learningowy „GX Works3 (Ladder)” (GX Works3 (drabinka))
- Instrukcje obsługi MELSOFT GX Works3

Struktura i składania programu

- Instrukcje obsługi MELSEC serii iQ-R

Po zakończeniu wszystkich etapów kursu **PLC Skuteczne programowanie**, możesz teraz przystąpić do testu końcowego. W razie niejasności w zakresie któregoś z tematów, wykorzystaj tę możliwość do ponownego zapoznania się z tymi zagadnieniami.

Test końcowy składa się z 9 pytań (10 elementów).

Możesz zdawać test końcowy dowolną ilość razy.

Jak rozwiązywać test

Po wybraniu odpowiedzi upewnij się, że przycisk **Odpowiedź** został kliknięty. Twoja odpowiedź zostanie utracona, jeśli będziesz kontynuować bez kliknięcia przycisku Odpowiedź. (Zostanie potraktowana jako pytanie, na które nie udzielono odpowiedzi.)

Punktacja końcowa

Liczba prawidłowych odpowiedzi, liczba pytań, procent prawidłowych odpowiedzi i wynik zaliczony/niezaliczony pojawią się na stronie wyniku.

Prawidłowe odpowiedzi: 4

Wszystkie pytania: 4

Procent prawidłowych odpowiedzi: 100%

Aby zaliczyć test musisz odpowiedzieć poprawnie na **60%** pytań.

Kontynuuj

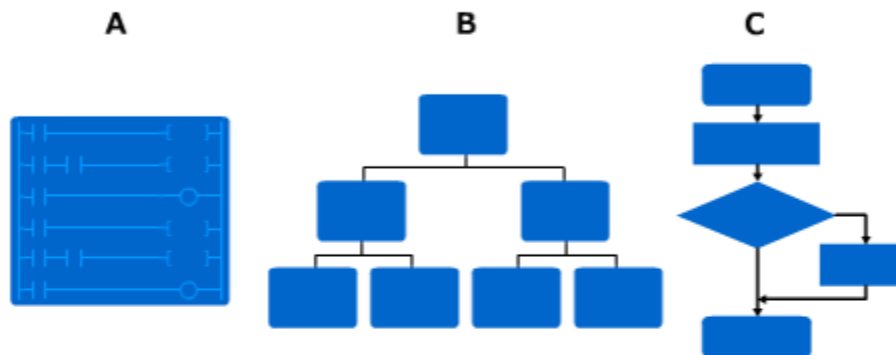
Przeglądaj

- Kliknij przycisk **Kontynuuj**, aby zakończyć test.
- Kliknij przycisk **Przeglądaj**, aby przeglądać test. (Sprawdzenie prawidłowych odpowiedzi)
- Kliknij przycisk **Spróbuj ponownie**, aby powtórzyć test.

Programy strukturalne

Wybierz prawidłową ilustrację programów strukturalnych.

- A
- B
- C



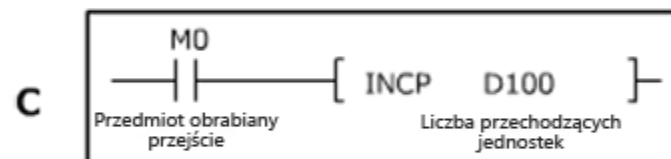
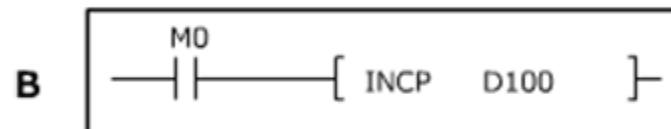
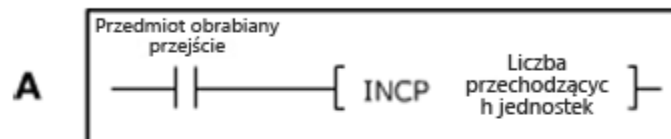
Odpowiedź

Wstecz

Wykorzystanie etykiet

Wybierz program utworzony przy użyciu etykiet.

- A
- B
- C



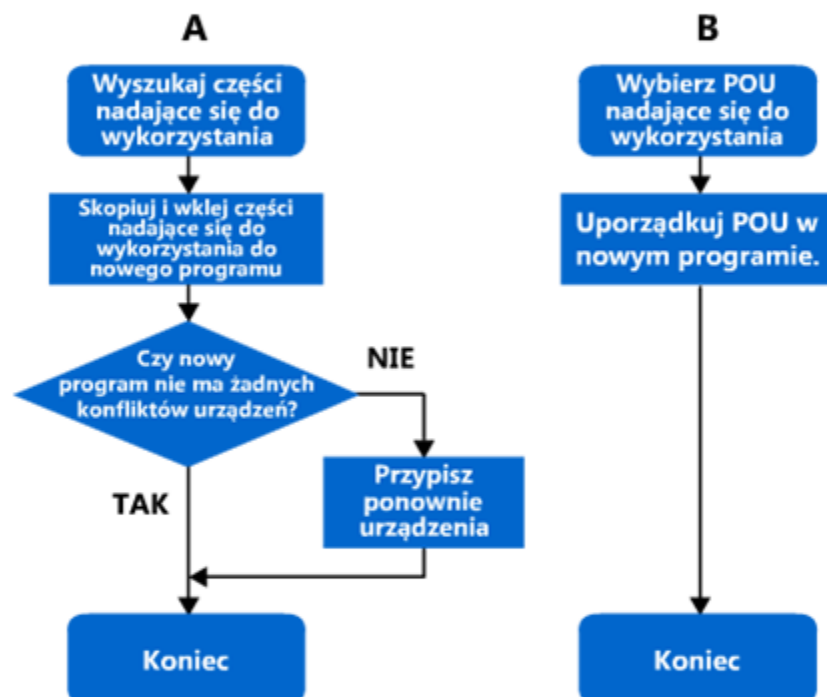
Odpowiedź

Wstecz

Proces wykorzystania programów

Wybierz wykres, który prawidłowo przedstawia proces wykorzystania programów strukturalnych utworzonych przy użyciu etykiet.

- A
- B



Odpowiedź

Wstecz

Cel programów strukturalnych

Wybierz prawidłowy powód stosowania programów strukturalnych. (Odpowiedzi wielokrotnego wyboru)

- Poprawia prędkość przetwarzania programu
- Poprawia skuteczność programowania
- Zmniejsza czas kompilacji programu
- Poprawia wizualizację programów

Odpowiedź

Wstecz

Programowanie strukturalne przy użyciu GX Works3

Wybierz prawidłowy opis dotyczący tworzenia programów strukturalnych w GX Works3.

- Programy strukturalne można tworzyć natychmiast po zainstalowaniu oprogramowania.
- Tworzenie programów strukturalnych w GX Works3 wymaga zakupu wtyczki.

Odpowiedź

Wstecz

Podział programów na części

Wybierz wszystkie stwierdzenia, które opisują podział programów na części. (Odpowiedzi wielokrotnego wyboru)

- Podział programów na wcześniej zdefiniowaną, równą liczbę kroków
- Podział programów na pojedyncze przetwarzanie i funkcje
- Przypisanie sensownych nazw do procesów
- Stosowanie wskaźników do przeskakiwania do różnych części
- Jednostki organizacyjne programu (POU) odnoszą się do wyniku podziału programu na części

Odpowiedź

Wstecz

Zalety bibliotek

Wybierz prawidłowy opis zalet rejestracji programów w bibliotekach. (Odpowiedzi wielokrotnego wyboru)

- Rejestracja często stosowanych POU umożliwia skuteczne wykorzystanie programów
- Zapobiega wykorzystaniu POU
- POU można udostępniać między wieloma osobami
- Rejestracja i ponowne wykorzystanie ujednoliconych programów zapewnia spójną jakość programu

Odpowiedź

Wstecz

Wykorzystanie bibliotek FB modułu i MELSOFT Library (biblioteki MELSOFT)

Wybierz prawidłowy opis wykorzystania bibliotek, takich jak blok funkcyjny modułu oraz MELSOFT Library.

- Nie ma konieczności weryfikacji operacji wewnętrznej w przypadku stosowania wstępnie opracowanych bibliotek
- FB modułu musi być utworzony zgodnie z modelem modułu

Odpowiedź

Wstecz

Wstępnie skonfigurowane biblioteki

Wybierz opcję, która odzwierciedla bibliotekę zapewnioną przez Mitsubishi Electric.

Q1 FB modułu

Q2 MELSOFT Library

Q1

Q2

Odpowiedź

Wstecz

Test końcowy został zakończony. Twoje wyniki są przedstawione poniżej.
Aby zakończyć test końcowy, przejdź do następnej strony.

Prawidłowe odpowiedzi: **9**

Wszystkie pytania: **9**

Procent prawidłowych odpowiedzi: **100%**

Kontynuuj

Przeglądaj

Gratulacje. Test został zaliczony.

Ukończyłeś(-aś) kurs **PLC Skuteczne programowanie**.

Dziękujemy za wzięcie udziału w kursie.

Mamy nadzieję, że poruszone tematy były interesujące, a informacje uzyskane w trakcie tego kursu będą przydatne w przyszłości.

Możesz przeglądać kurs dowolną ilość razy.

Przełdaj

Zamknij